



Open your mind. LUT.
Lappeenranta University of Technology

**TRUKIN LIIKKEEN TUNNISTAVAN SULAUTETUN
JÄRJESTELMÄN TOTEUTUS**
**The implementation of a motion-sensitive embedded
system for a forklift**

Teemu Öster

TIIVISTELMÄ

Lappeenrannan teknillinen yliopisto
Teknillinen tiedekunta
Sähkötekniikan koulutusohjelma

Teemu Öster

Trukin liikkeen tunnistavan sulautetun järjestelmän toteutus

2012

Kandidaatintyö.
30 sivua, 17 kuvaa

Tarkastaja: professori Jero Ahola

Trukeissa voidaan nykyään käyttää tietokoneita tehostamassa kuljettajien työskentelyä, mutta koneet aiheuttavat myös työturvallisuusriskin, jos niitä käytetään ajaessa. Tässä kandidaatintyössä suunnitellaan ja toteutetaan prototyyppi laitteesta, joka sammuttaa trukin näytön kun se liikkuu. Liikkeen havaitseminen toteutetaan käyttämällä analogisia kulmanopeus- ja kiihtyvyyssantureita. Anturien signaaleja luetaan ja suodatetaan Arduino Uno-mikrokontrollerikehitysalustaa käyttämällä. Mikrokontrollerilla ohjataan kytkimenä käytettävän transistoria. Transistori kytkee tietokoneen näytön taustavalon pois päältä kun trukki liikkuu.

Laitteen testaus suoritettiin henkilöautolla ja näytön paikalla käytettiin summeria. Tavoittein päästiin muuten paitsi nopeuden laskemisen osalta, jota tarvitaan liiketunnistukseen, kun trukki liikkuu suoraan tasaista nopeutta. Tämä johtuu kiihtyvyyssanturin epätarkkuudesta. Testeissä huomattiin kuitenkin että liiketunnistinta voidaan pitää toimivana, koska tunnistusvirhe ilmenee vain silloin kun ajoalusta on erittäin tasainen ja nopeus miltei vakio.

ABSTRACT

Lappeenranta University of Technology
Faculty of Technology
Degree Programme in Electrical Engineering

Teemu Öster

The implementation of a motion-sensitive embedded system for a forklift

2012

Bachelor's Thesis.
30 pages, 17 pictures

Examiner: professor Jero Ahola

Nowadays computers can be used in forklifts to make the working of the drivers more efficient. These computers can cause safety risks if the drivers use them while driving. In this Bachelor's Thesis a prototype of a device, which shuts down the forklift's computer's screen while it is moving, is designed and implemented. Movement is detected by using an analog accelerometer and gyroscope. The outputs of the sensors are read and filtered with Arduino Uno microcontroller prototyping platform. The microcontroller controls a transistor that operates as a switch. The transistor turns the backlight of the computer's screen off while the forklift is moving.

The testing of the device was made with a car and a buzzer was used instead of a screen. Goals of Bachelor's Thesis were reached otherwise, except the calculation of speed that is needed in motion detecting when forklift is moving straight with constant speed. This is due to the fact that the accelerometer is not ideal. In tests however was found that the motion detector can be considered as functional because the error occurs only when the driving platform is flat and the speed is almost constant.

SISÄLLYSLUETTELO

KÄYTETYT MERKINNÄT JA LYHENTEET	5
1. Johdanto	7
2. Laitteisto ja kytkentä.....	9
2.1 Anturit.....	9
2.1.1 Kiihtyvyyssanturi.....	10
2.1.2 Kulmanopeusanturi.....	11
2.2 Arduino.....	12
2.3 Kytkentä	13
3. A/D-muunnos ja suodattimen suunnittelu	16
3.1 A/D-Muunnos	16
3.2 Suodattimen suunnittelu	19
3.3 Nopeuden laskeminen.....	20
4. Ohjelmoitu logiikka.....	23
5. Testit ja mittaukset	25
6. Yhteenveto.....	29
LÄHTEET	30

KÄYTETYT MERKINNÄT JA LYHENTEET

ASIC	Application-specific integrated circuit
MEMS	Micro Electro Mechanical Systems
RAM	Random Access Memory
USB	Universal Serial Bus
<i>a</i>	Kiihtyvyys
<i>A</i>	Pinta-ala
<i>b</i>	Bitti
<i>C</i>	Kapasitanssi
<i>D</i>	Kondensaattorien levyjen välinen etäisyys
<i>e</i>	Virhe
<i>f</i>	Taajuus
<i>I</i>	Kollektorivirta kyllästystilassa
<i>N</i>	Näytteiden määrä
<i>Q</i>	Kvantisointitaso
<i>R</i>	Kollektori resistanssi
<i>t</i>	Aika
<i>v</i>	Nopeus
<i>V</i>	Jännite
<i>w</i>	Kulmanopeus
<i>x</i>	Näyte
ΔV	Jännite-ero
β	Virtavahvistuserroin
ε	Dielektrinen vakio

Alaindeksit

acc	Kiihtyvyyssanturin signaali
accSum	x-, y- ja z-akselin suuntaisten kiihtyvyyksien summavektori
accx	Kiihtyvyyssanturin x-akselin suuntaisen kiihtyvyyden signaali
accy	Kiihtyvyyssanturin y-akselin suuntaisen kiihtyvyyden signaali
accz	Kiihtyvyyssanturin z-akselin suuntaisen kiihtyvyyden signaali
ave	Keskiarvo
B	Kanta
B(EOS)	Kanta kyllästystilan rajalla
BEsat	Kanta – emitteri kyllästystilassa

C	Kollektori
CC	Käyttöjännite
Csat	Kollektori kyllästystilassa
CEsat	Kollektori – emitteri kyllästystilassa
gyro	Kulmanopeusanturin signaali
gyrox	Kulmanopeusanturin x-akselin ympäri olevan kulmanopeuden signaali
gyroy	Kulmanopeusanturin y-akselin ympäri olevan kulmanopeuden signaali
MAX	Maksimi
Offset	Arvo, joka lisätään mitattuun arvoon, jotta paikallaan ollessa näyttää oikein
q	Kvantisointi
Q_m	Kvantisointitasojen määrä
$Q - 1$	Ylin kvantisointitaso
s	Näytteistys
v_0	Nopeuden ollessa 0

1. JOHDANTO

Trukeissa käytetään nykyään tietokoneita apuna tehostamaan kuljettajan työskentelyä. Kuljettajat saavat tietokoneiden kautta tehtävät, mitä siirtää minnekin ja että missä mitäkin on. Nämä tietokoneet aiheuttavat työturvallisuusrisikin, jos ajaja käyttää konettaan ajaessaan, joten koneen käyttö pitäisi pystyä estämään silloin kun trukki liikkuu. Työssä tavoitteena on suunnitella kiihtyvyy- ja kulmanopeusanturia käyttämällä laite, joka tunnistaa milloin ollaan liikkeessä, jolloin trukin tietokoneen näyttö saadaan sammutettua ja kytkettyä takaisin päälle, kun pysähdytään. Kuva trukista, sekä sen näytöstä ja näppäimistöä on esitetty kuvassa 1.1.

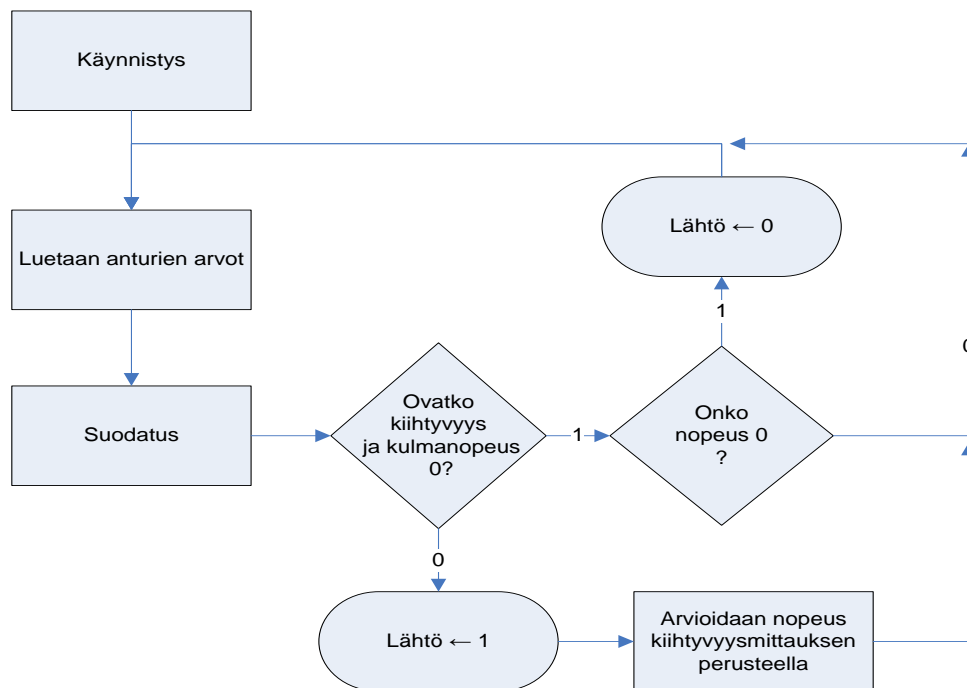


Kuva 1.1. Vasemmalla puolella on kuva trukista johon ensimmäinen laite asennettiin ja oikealla on kuva trukin tietokoneen näytöstä ja näppäimistöä.

Projektia aloitettaessa ehtona oli, että liiketunnistimen pitää olla helposti asennettavissa ja sen pitää käydä erimallisiin trukkeihin. Tämän takia liiketunnistusta ei alettu suunnitella trukin mekaanisiin osiin kuten akseleihin. Liiketunnistuksen toimivuutta kulmanopeus- ja kiihtyvyyssantureilla testattiin ensin älypuhelimella, joka sisältää kyseiset anturit ja havaittiin, että toimivan liiketunnistimen tekeminen näillä antureilla on mahdollista. Liiketunnistimen tekemiseen tarvitaan antureiden lisäksi mikrokontrolleri, jolla voidaan lukea antureita ja johon ohjelmoidulla logiikalla voidaan antureilta tulevaa dataa suodattaa. Trukin tietokoneen näytössä on digitaalinen sisääntulo, jolla näytön taustavalo saadaan sammutettua, joten mikrokontrollerissa täytyy olla digitaalinen lähtö, jota logiikalla pystytään ohjaamaan.

Suodattimen suunnittelussa on tavoitteena saada moottorin tärinästä aiheutuvat kiihtyvyydet ja kulmanopeudet suodatettua pois, jotta trukin ollessa paikallaan käynnissä, näyttöä ei suljettaisi. Tarvitaan siis alipäästösuodin. Toisaalta suodattimen rajataajuus ei saa olla liian pieni, jotta mahdollisimman pienet muutokset trukin liikkeessä havaitaan. Ongelmana on tilanne, jossa trukki liikkuu suoraan tasaisella alustalla tasaista nopeutta, eli kiihtyvyyden sekä kulmanopeuden komponentit ovat nolla. Tätä varten tarvitsee laskea kiihtyvyyden integraali, jolloin tiedetään nopeus.

Työssä suunnitellaan prototyyppi, joka ei sisällä mikroprosessorin lähdön ja trukin näytön välistä elektroniikkaa, koska prototyypin suunnittelu ja testaus vaiheessa ei ollut käytettävissä truckia. Prototyypissä on näytön paikalla summeri, jota ohjataan samalla logiikalla ja digitaalisella lähdöllä, kuin tuotantoversiossa näyttöä. Suunnitelmakaavio liiketunnistimen toiminnasta on esitetty kuvassa 1.2.

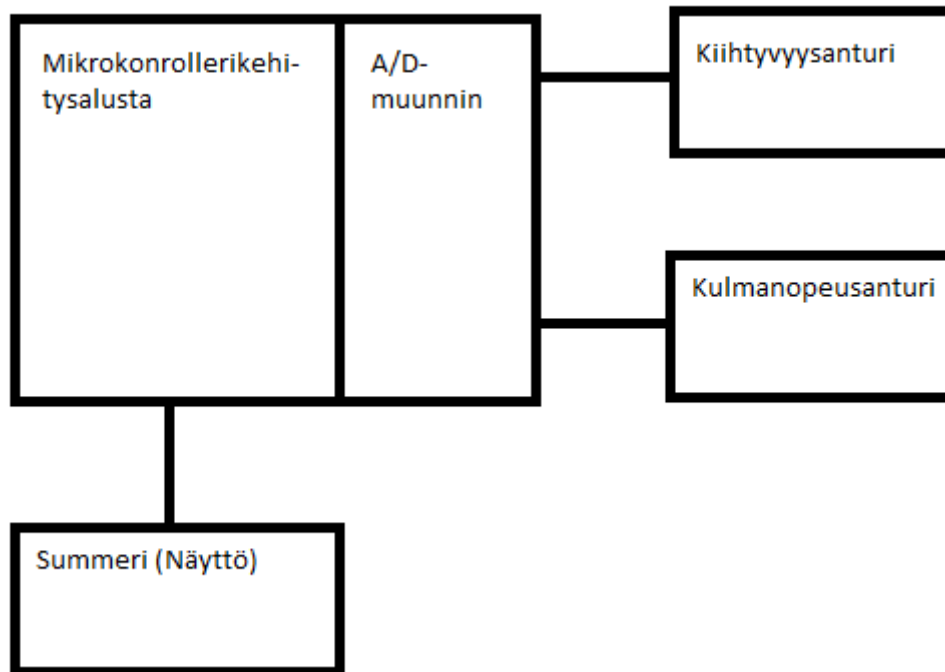


Kuva 1.2. Suunnitelma liiketunnistimen toteutusta varten. Lähdön ollessa 0 ollaan paikallaan ja kun lähtö on 1 liikutaan.

Raportin toisessa kappaleessa käsitellään liiketunnistimen laitteistoa, sekä kytkentää. Kolmannessa kappaleessa kerrotaan suodattimen suunnittelusta ja tarkastellaan A/D muunnosta. Seuraavaksi esitetään mikrokontrolleriin ohjelmoitu logiikka ja tämän jälkeen testataan laitteen toimivuutta. Yhteenvedossa pohditaan projektin onnistumista, sekä mietitään miten liiketunnistin saataisiin paremmaksi.

2. LAITTEISTO JA KYTKENTÄ

Kuvassa 2.1 on esitetty lohkokaaviokuva toteutettavasta laitteistosta.



Kuva 2.1. Lohkokaaviokuva liiketunnistimen laitteiston rakenteesta. Anturien ollessa analogiset ne ovat kytketty mikrokonrollerikehitys-alustan A/D-muunttimeen. Näytön paikalla prototyypissä oleva summeri on kytketty digitaaliseen lähtöön.

Laitteiston toiminta käydään läpi alkaen antureista ja päättyen mikrokonrollerilla ohjattuun summereihin. Kappaleen lopussa esitetään liiketunnistimen kytkentä.

2.1 Anturit

Valittaessa antureita prioriteetteina oli toimivuus, hinta, helppokäyttöisyys ja saatavuus. Kiihtyvyyssanturissa tulee olla useamman akselin suuntainen mittaus kiihdyttämisen/jarruttamisen sekä trukin ohjaamisen lisäksi, joita tarvitaan nopeuden määrittämiseksi epätasaisella alustalla. Antureiden tulee myös olla mittausalueeltaan, sekä tarkkuudeltaan trukin liikkeeseen sopivia. Kiihtyvyyden mittausalueeksi oletetaan riittävän ± 1 g eli $9,81$ m/s^2 ja kulmanopeuden $\pm 100^\circ/s$. Työssä käytetään liitännältään analogisia antureita, koska ne ovat hinnaltaan digitaalisia edullisempia.

Käytettävät anturit perustuvat MEMS-tekniikkaan (Micro Electro Mechanical Systems). MEMS komponentit ovat kooltaan muutamasta mikrometrinä muutamaan sataan mikro-

metriin, ja ne voidaan jakaa kahteen kategoriaan, antureihin ja toimilaitteisiin. MEMS-anturit ovat laitteita, jotka havaitsevat ympäristönsä tilan ja tuottavat vastaavan lähtösignaalin. [1]

2.1.1 Kiihtyvyyssanturi

Kiihtyvyyssanturiksi valittiin kolmi-akselinen analoginen kiihtyvyyssanturi MMA7316. MMA7316 on kapasitiivinen MEMS-anturi, jossa kiihtyvyys havaitaan mittaamalla kapasitanssin muutosta liikkuvan ja paikallaan olevien elektrodien avulla. Anturi sisältää mittauselementin, sekä signaalin muokkaus ASIC:in (Application-specific integrated circuit). [2] Toimintaperiaate on havainnollistettu kuvassa 2.2.

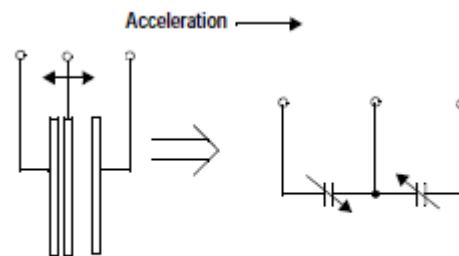


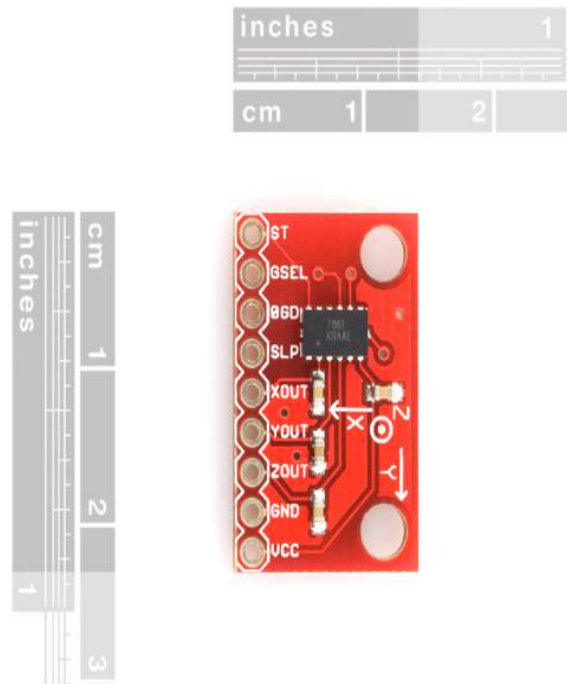
Figure 3. Simplified Transducer Physical Model

Kuva 2.2. MMA7316 kiihtyvyyssanturin toimintaperiaate yksinkertaistettuna. Kun anturi on oikealle päin suuntautuvassa kiihtyvässä liikkeessä, pienenee keskimmäisen ja vasemman puoleisen levyn etäisyys, joka johtaa kapasitanssin muutokseen levyjen välillä ($C = A\epsilon / D$). ASIC käyttää ns. switched capacitor – tekniikkaa mittauselementin kapasitanssien mittaukseen, sekä kiihtyvyyssanan keräämiseen näiden kahden kondensaattorin kapasitanssien eroista. ASIC myös vahvistaa ja suodattaa signaalia, jonka jälkeen lähtöön saadaan signaali joka on verrannollinen kiihtyvyyteen.[3]

Anturi on valmiiksi piirilevyllä, jossa on lähtösignaaleille alipäästösuoittimet ja suodatuskondensaattori käyttöjännitteelle, joka on esitetty kuvassa 2.3. Anturi toimii 2,2 – 3,3 V käyttöjännitteellä. Anturin lähtöimpedanssi on 32 k Ω ja suodatus on toteutettu kytkemällä 2,2 nF kondensaattori jokaisen lähdön ja maan välille. [3], [4]

Anturissa on kaksi eri mittausaluetta valittavissa [3]. Valinta tehdään anturin piirilevyn GSEL-pinnistä. Vaihtoehdot ovat $\pm 1,5$ g ja ± 6 g, eli noin 14,7 m/s² ja 88,3 m/s² [3]. Valittiin mittausalueeksi $\pm 1,5$ g, koska truckki ei saavuta suurempia kiihtyvyyksiä ja koska siten saatiin parempi tarkkuus, joka on 800 mV/g [3]. Lähdön arvot vaihtelevat välillä 0,45 – 2,85 V ja kiihtyvyyden ollessa 0 m/s² lähtöjännite on 1,65 V [3]. Nyt GSEL voitiin jättää kytkemättä, jolloin saatiin haluttu mittausalue [3]. Self test (ST-pinni), sekä 0g-Detect (0GD-pinni) jätettiin kytkemättä, sillä niitä ei tässä sovelluksessa tarvita. SLP-pinni sen sijaan kytketään mikrokontrollerin digitaaliseen lähtöön. Kun siihen syötetään 0, anturi

menee sleep modeen, jolloin anturien lähdöt menevät nolnaan [3]. Eli SLP-pinniin on syötettävä 1 aina laitteen ollessa toiminnassa [3].



Kuva 2.3. Kiihtyvyyssanturi MMA7361 piirilevyllä. Anturi asennetaan siten, että kuvassa näkyvä akselin x-nuoli osoittaa kiihtyvyyttä eteenpäin, akselin y-nuoli sivulle ja akselin z nuoli ylöspäin. [4]

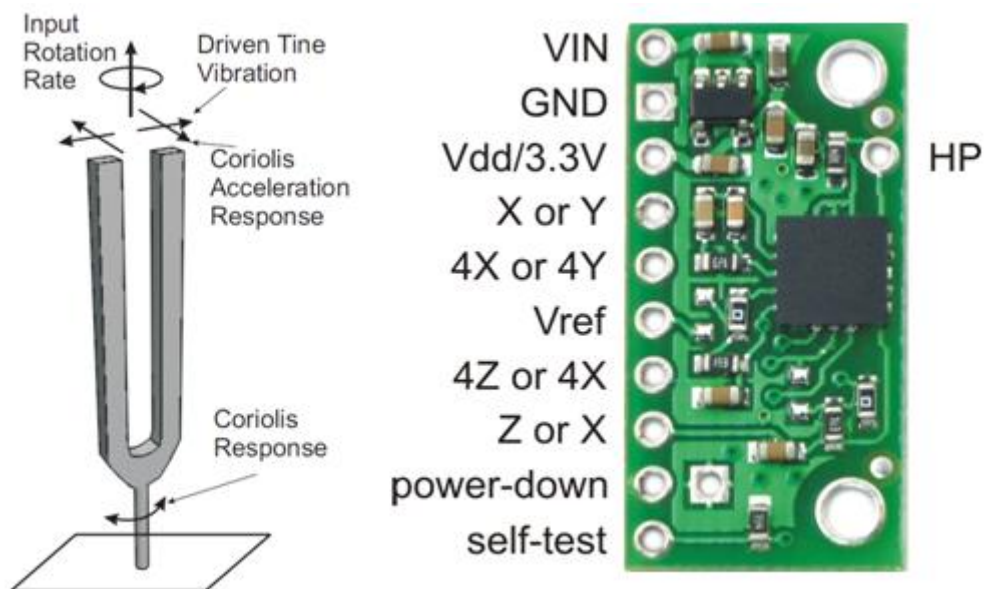
2.1.2 Kulmanopeusanturi

Trukin kääntymistä ja sivuttaisheilahduksia mittaavaksi kulmanopeusanturiksi valittiin analoginen kaksiakselinen LPR510AL. MEMS-kulmanopeusanturit ovat yleensä värähtelevärakenteisia, kuten myös LPR510AL [5],[6]. Värähtelevärakenteiset gyroskoopit, kuten kulmanopeusantureita myös kutsutaan, toimivat siten että ne laitetaan värähtelemään, ja kun anturi pyörii, aiheuttaa pyörimisestä johtuva coriolis-kiihtyvyyks muutosta värähtelyssä [6]. Tätä voidaan mitata esimerkiksi kapasitiivisesti, jolloin saadaan kulmanopeudella moduloitu signaali [7]. Kuvassa 2.4 on esitetty yksi värähtelevärakenteinen gyroskooppi, jota kutsutaan tuning fork-gyroskoopiksi. Värähtelevärakenteisia gyroskooppeja on myös muilla tavoilla toteutettuja, mutta toimintaperiaate on niissä kaikissa kuitenkin sama. LPR510AL – datalehti [5] ei kerro, että minkälainen sisäinen värähtelevä rakenne anturissa on.

Piirilevy, johon anturi on integroitu, sisältää jänniteregulaattorin, jolloin sitä voidaan käyttää 2,7 - 16 V jännitteillä. Syöttö tapahtuu 2,7 – 3,6 V tapauksessa Vdd/3.3V-pinniin ja korkeammilla jännitteillä Vin-pinniin, jolloin käytetään regulaattoria jännitteen alentamiseksi anturille sopivaksi. Levyllä on myös suodatuskondensaattoreita, sekä alipäästösuodat-

timet lähdöille joiden rajataajuus on 140 Hz. [8] Anturi ja sen piirilevy on esitetty kuvassa 2.4.

Anturin akselien mitta-alueet on valittavissa, joko $\pm 100^\circ/\text{s}$, tai $\pm 400^\circ/\text{s}$ [5], [8]. Molemmilla akseleilla on kaksi lähtöä, joista toinen on vahvistettu nelinkertaiseksi ja sen herkkyys on $10 \text{ mV}/^\circ/\text{s}$ [5], [8]. Käytetään vahvistettua lähtöä, jolloin saadaan parempi tarkkuus. Lähtöjen jännitteet ovat 1,23 V, kun ollaan paikallaan, jolloin lähtöjen arvot vaihtelevat välillä 0,23 – 2,23 V [5], [8]. Nollatason jännite on valittavissa Vref-pinnistä [5], [8], mutta nyt sitä ei tarvitse muuttaa, joten pinni voidaan jättää kytkemättä. Power-down- ja self-test-pinnit kytketään maahan sillä näitä ominaisuuksia ei käytetä.

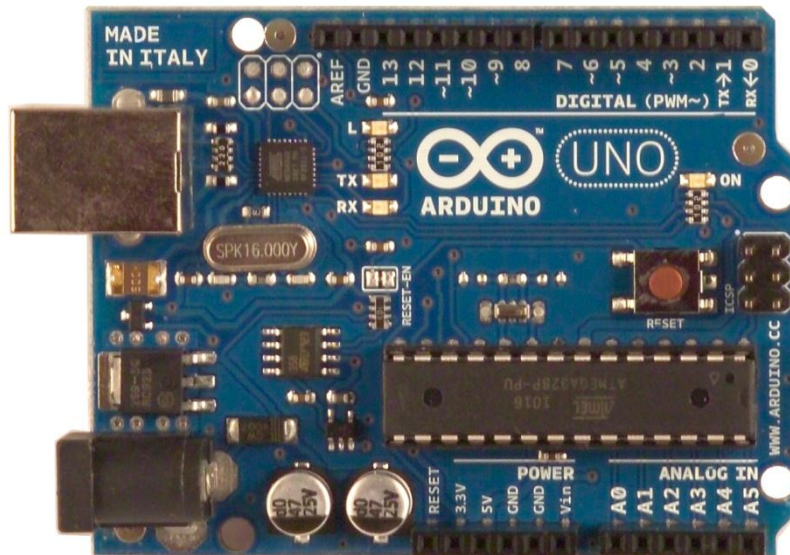


Kuva 2.4. Vasemmalla tuning fork-gyroskoopin toiminta periaate [6] ja oikealla kulmanopeusanturi LPR510AL piirilevyllä [8]. Anturi asennetaan siten, että kuvassa näkyvä alempi vahvistettu lähtö mittaa trukin ohjaamista ja ylempi ohjauskulman pystysuuntaista muutosta.

2.2 Arduino

Mikrokontrollerialustaksi valittuun Arduinoon päädyttiin aikaisempien käyttökokemusten perusteella. Hinta ja soveltuvuus tähän projektiin puolsivat myös Arduinoon käyttöä. Arduino on avoimeen lähdekoodiin perustuva mikrokontrollerikehitysalusta. Ohjelmointiympäristö on helppokäyttöiseksi suunniteltu, ohjelmoijan ei tarvitse puuttua mikrokontrollerin rekistereihin ja ohjelmointiin on saatavilla työkalu Arduino Ide ilmaiseksi Arduinoon kotisivuilta. [9] Valittaessa Arduinoon kehitysalustoista sopivaa, tärkeitä ominaisuuksia on kellotaajuus, joka on mallista riippuen 8 – 16 MHz, analogisten tulojen määrä, joita tarvitaan viisi, A/D-muuntimen bittien määrä, joka on kaikissa Arduinoissa 10, RAM:in (Random Access Me-

mory) määrä, jota tulisi olla vähintään 1 kilotavu, sekä hinta. Valitaan Arduino Uno, siinä on 16 MHz kellotaajuus, 6 analogista tuloa, 10-bittinen A/D-muunnin, 2 kilotavua RAM:ia, sekä se on hinnaltaan yksi halvimmista Arduinoista [9]. Arduino Uno piirilevyn päältä kuvattuna on esitetty kuvassa 2.5.



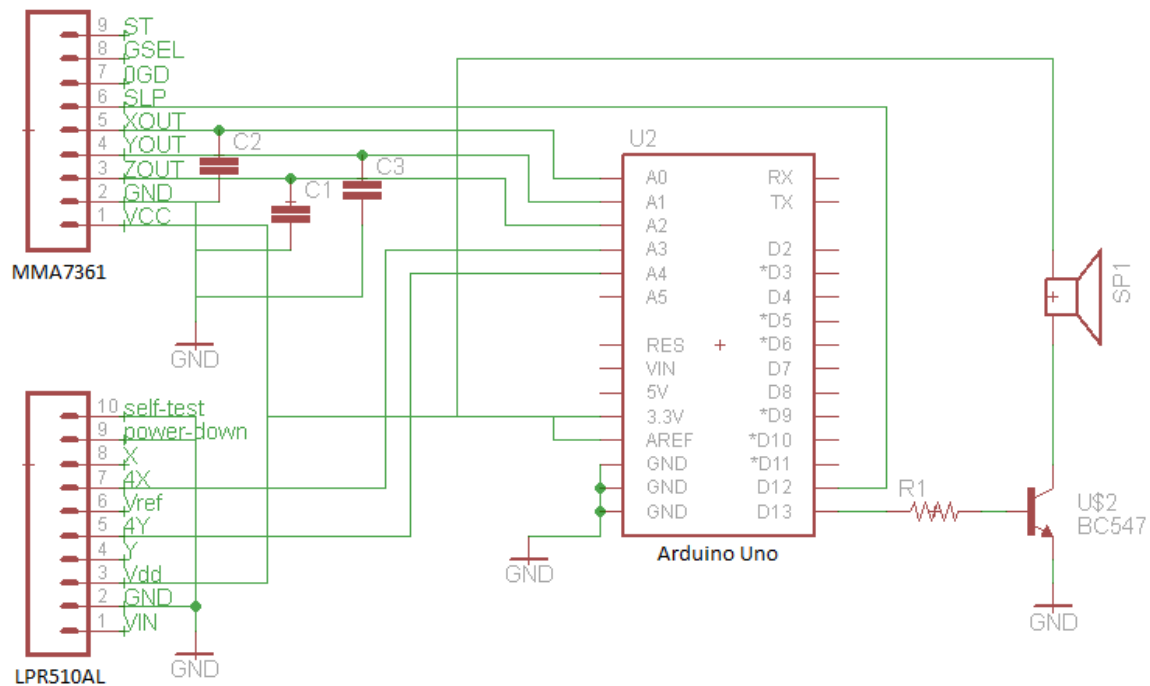
Kuva 2.5. Mikrokontrollerikehitysalusta Arduino Uno. Mikrokontrollerina toimii ATmega 328. Analogiset sisääntulot ovat levyn oikeassa alalaidassa. [9]

Arduinoa ohjelmoidaan C-kielellä, Ohjelman lataus mikrokontrollerille tapahtuu USB-kaapelilla (Universal Serial Bus), josta levy saa samalla myös käyttösähköt. USB-portti näkyy kuvassa 2.5 levyn oikeassa yläkulmassa. USB-sarjajunnoksen hoitaa ATmega 8U2 mikrokontrolleri. Arduino Uno:n toimintajännite on 5 V. Jännite voidaan syöttää USB-liitännän kautta, tai ulkoisesta jännitelähteestä. Ulkoista lähdettä käytettäessä suositeltu jännite on 7 – 12 V. Uno osaa valita automaattisesti käytettävän jännitelähteen. [9] Ylärivissä on digitaalisten tulo/lähtö-pinnien lisäksi yksi maa-pinni, sekä AREF pinni, johon syöttämällä jännitettä voidaan valita referenssijännite. Tämän avulla voidaan säätää A/D-muunnoksen tarkkuutta, sillä referenssijännite on maksimijännite, jolloin digitaaliseksi muunnetun arvon kaikki 10 bittiä ovat ykkösiä, eli arvo on 1023 [9]. Oletuksena arvona on 5 V [9], mutta koska antureilta tulevat jännitteet ovat maksimissaan 2,85 V kytketään referenssijännitteeksi Arduino-levyllä olevasta pinnistä 3,3 V. A/D-muunnoksesta tarkemmin kolmannessa kappaleessa.

2.3 Kytkeä

Liiketunnistimen kytkentä on esitetty kuvassa 2.6. Kiihtyvyyssanturin mittaussignaalien lähdöt on kytketty Arduinon analogisiin tuloihin siten, että kiihtyvyys eteen – taakse on kytket-

ty tuloon A0, sivuttaissuuntainen kiihtyvyys on kytketty tuloon A1, ja pystysuuntainen tuloon A2. Kulmanopeusanturin trukin kääntymistä mittaavan anturin lähtö on kytketty tuloon A3 ja pystysuuntaisen ohjauskulman muutosta mittaava lähtö tuloon A4. Koska oletetaan että halutaan mitata taajuudeltaan noin ≤ 10 Hz olevia liikkeitä, lisätään kiihtyvyysanturin mittaussignaalien lähtöihin kondensaattorit. Näin saadaan alipäästösuodattimen rajataajuus pienemmäksi ja samalla kohinatehoa vähennettyä. Kondensaattorien mitoitus tehdään seuraavassa kappaleessa.



Kuva 2.6. Liiketunnistimen kytkentä. Käyttösähköt tuodaan Arduinolle liittimestä, jota ei ole kuvassa.

Arduinon 3,3 V-pinnistä, josta otetaan A/D-muunnoksen referenssijännite, otetaan myös käyttösähköt antureille, sekä summerille. Summeria ohjataan Arduinon digitaalisella ulostulolla D13. Kun mikrokontrolleriin ohjelmoitu logiikka havaitsee anturien tuottamien mittaussignaalien perusteella että trucki liikkuu, D13 pinnistä syötetään 5 V vastuksen R_1 läpi kytkimenä toimivan BJT transistorin (BC547B) kantaan. $R_1 = R_B$ mitoitetaan siten, että kun pinnan D13 jännite nousee $0 \rightarrow 5$ V, transistori menee kyllästystilaan.

Summerin resistanssi määritettiin mittaamalla jännite summerin yli, joka oli 3,25 V ja summerin läpi kulkeva virta, joka oli 13 mA. Näistä saadaan laskettua summerin resistanssi Ohmin lakia käyttämällä $R_C = 250 \Omega$. Kollektori-emitterijännite kyllästystilassa on $V_{CEsat} = 0,2$ V [10]. Lasketaan kollektorivirta kyllästystilassa yhtälön (2.1) avulla [11].

$$I_{C\text{sat}} = \frac{V_{CC} - V_{CE\text{sat}}}{R_C} = 12,4 \text{ mA} \quad (2.1)$$

V_{CC} on summerin käyttöspankki 3,3 V. Nyt voidaan laskea pienin kantavirta I_B , joka tarvitaan, jotta saadaan transistori kyllästystilaan. Virtavahvistuskertoimen on datalehden mukaan $\beta = 200 - 450$ [10]. Minimi kantavirran arvo, jolla transistori menee kyllästystilaan, on laskettu yhtälössä (2.2). Laskussa käytetään virtavahvistuskertoimen minimiarvoa. [11]

$$I_{B(\text{EOS})} = \frac{I_{C\text{sat}}}{\beta_{\text{min}}} = 0,062 \text{ mA} \quad (2.2)$$

Kanta-emitterijännite on kyllästystilassa $V_{BE\text{sat}} = 0,7 \text{ V}$ [10]. Lasketaan yhtälöä (2.3) käyttäen vastuksen R_B arvo [11].

$$R_{B(\text{EOS})} = \frac{+5 \text{ V} - V_{BE\text{sat}}}{I_{B(\text{EOS})}} = 69,4 \text{ k}\Omega \quad (2.3)$$

Tämä on suurin vastuksen arvo, jota voidaan käyttää, jotta transistori menee kyllästystilaan. Kantavirran kasvattamisella ei ole suurta vaikutusta kollektorivirtaan kyllästystilassa, joten vastuksen R_B arvoksi valitaan 5,1 k Ω , jolloin kantavirta I_B yhtälöstä (2.3) ratkaistuna on 0,84 mA. [11]

3. A/D-MUUNNOS JA SUODATTIMEN SUUNNITTELU

Antureilta tuleva mittausdata on analogista ja se täytyy muuntaa digitaaliseksi, jotta dataa pystytään käsittelemään mikrokontrollerilla. Anturit sisältävät analogiset alipäästösuodattimet, mutta mittausdataa täytyy suodattaa myös digitaalisesti, jotta kohinaa saadaan pienennettyä.

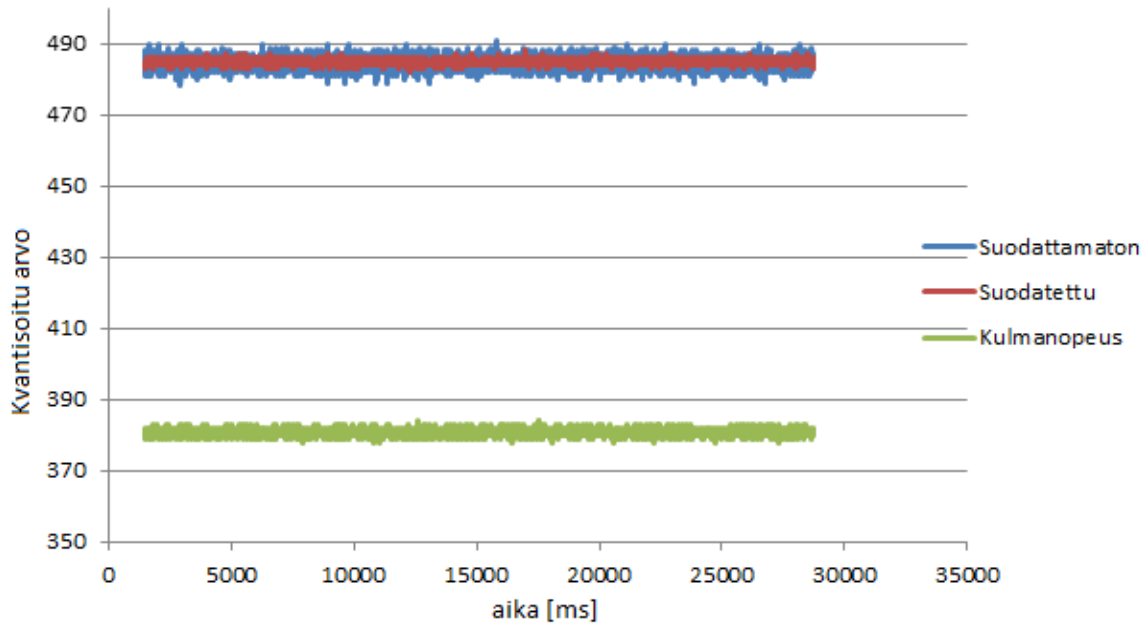
3.1 A/D-Muunnos

Analogisen signaalin muuntaminen digitaaliseksi tapahtuu neljässä vaiheessa, jotka ovat suodatus, näytteistys, kvantisointi ja koodaus.

Analogisen signaalin suodatus tapahtuu kulmanopeusanturissa sen piirilevyllä valmiiksi olevilla alipäästösuodattimilla, joiden rajataajuus on 140 Hz [5], [8]. Kiihtyvyyssanturin suodattimen x- ja y-akselien lähtöjen -3dB:n rajataajuus 400 Hz ja z-akselin lähdön 300 Hz [2], mutta näitä signaaleja suodatetaan vielä erikseen analogisilla alipäästösuodattimilla, jotta niiden rajataajuus saadaan pienemmäksi ja kohinaa vähennettyä. Suodattamattomat kulmanopeusanturin kääntymistä mittaavan x-akselin signaali, sekä kiihtyvyyssanturin eteen – taakse kiihtyvyyttä mittaavan x-akselin signaali on esitetty kuvassa 3.1. Kuvassa on myös analogisesti suodatettu kiihtyvyyssanturin signaali.

Vaikka kappaleessa 2.3 sanotaan että halutaan mitata taajuudeltaan ≤ 10 Hz signaaleja, asetetaan kiihtyvyyssanturin signaalien suodattimien rajataajuudeksi 50 Hz. Rajataajuus jätetään korkeammaksi, koska signaaleja käsitellään vielä digitaalisesti. Kuvasta 3.1 nähdään, että kulmanopeusanturilta tulevat signaalit sisältävät vähemmän kohinaa, joten niitä ei erikseen suodateta analogisesti. Kuten kappaleessa 2.1.1 kerrotaan, kiihtyvyyssanturilla on 32 k Ω lähtöimpedanssi, sekä $C_{acc} = 2,2$ nF kondensaattori kytkettynä lähdön ja maan välille. Rajataajuutta voidaan laskea lisäämällä kapasitanssia, eli kytkemällä toinen kondensaattori anturin piirilevyllä olevan rinnalle. Yhtälössä (3.1) lasketaan tarvittava kapasitanssi C , jotta rajataajuudeksi saadaan 50 Hz.

$$C = \frac{1}{2\pi f R} - C_{acc} = \frac{1}{2\pi * 50 \text{ Hz} * 32 \text{ k}\Omega} - 2,2 \text{ nF} = 97 \text{ nF} \quad (3.1)$$



Kuva 3.1. Kiihtyvyyssanturin x-akselilta tulevan signaalin kvantisoidut arvot ennen analogista suodattamista ja sen jälkeen, sekä kulmanopeussanturin x-akselilta tulevan signaalin kvantisoidut arvot. Kiihtyvyyssanturin signaali saa arvoja ennen suodatusta väliltä 478 – 491 ja suodatuksen jälkeen 482 – 488 keskiarvon ollessa 485. Kulmanopeussanturilta tulevat arvot vaihtelevat välillä 378 – 384 keskiarvon ollessa 381. Anturi oli mittaushetkellä paikallaan vaakatasossa.

Näytteistyksessä oleellinen asia on näytteistystaajuus f_s , joka on näytteenoton välisen ajan käänteisarvo. Nyquistin näytteenottoteoreeman mukaan alkuperäinen signaali voidaan rekonstruoida otetuista näytteistä, mikäli näytteenottotaajuus on vähintään kaksinkertainen alkuperäisen signaalin maksimitaajuuteen nähden. Jos näytteistystaajuus $f_s < 2 * f_{MAX}$, signaali laskostuu, eli rekonstruoitu signaali ei vastaa alkuperäistä. [12]

Seuraavaksi otetut näytteet pyöristetään joko ylös-, tai alaspäin seuraavalle kvantisointitasolle. Näiden tasojen määrä Q_m riippuu siitä, että kuin monta bittiä käytettävässä A/D-muuntimessa on. Peräkkäisten tasojen välinen jännite-ero ΔV voidaan laskea yhtälöä (3.2) käyttämällä. [12]

$$\Delta V = \frac{V_{Q_m-1} - V_0}{Q_m - 1} \quad (3.2)$$

Kuten kappaleessa 2.2 mainittiin, Arduino Unossa on 10 bittinen A/D-muunnin, jolloin kvantisointitasojen määrä $Q_m = 2^{10} = 1024$. Referenssijännitteeksi on asetettu $V_{ref} = 3,3$ V, eli $V_{Q-1} = V_{ref}$ ja $V_0 = 0$ V. Nyt kahden peräkkäisen tason välisen jännite-eron likiarvo on yhtälön (3.2) mukaisesti $\Delta V = 3,23$ mV. Kvantisoinnissa syntyy virhettä, joka johtuu pyöristämisestä [12]. Virhe on maksimissaan $e_q = \Delta V / 2$, eli tässä tapauksessa 1,61 mV [12].

A/D-muuntimelle tulevasta jännitteestä saadaan laskettua arvo, jonka se kvantisoinnissa saa yhtälöllä

$$Q_n = \frac{V}{\Delta V} \quad (3.3)$$

Kiihtyvyyssanturin jännitteet vaihtelevat välillä 0,45 – 2,85 V, jolloin kaikkia kvantisointitasoja ei käytetä, vaan anturilta tulevat signaalit voivat yhtälöllä (3.3) laskettuna saada kvantisoinnissa arvoja 139 – 883. Kiihtyvyyden ollessa 0 m/s² jännitteen tulisi olla 1,65 V ja tällöin kvantisointitason $Q_{\text{accx},v0}$ tulisi olla yhtälön (3.3) mukaan 511. Kuvasta 3.1 kuitenkin havaitaan, että näyttöiden liukuva keskiarvo on $Q_{\text{accx}} = 485$. Myös muiden kiihtyvyyssanturin akselien arvoissa on virhettä. Kulmanopeusanturilta mittauskanavien lähtöjännitteet saavat arvoja väliltä 0,23 – 2,23 V, eli kvantisoinnissa saatavat arvot yhtälöä (3.3) käyttämällä ovat 71 – 690. Kun kulmanopeus on $w = 0$ °/s, jännite on 1,23 V ja sitä vastaava kvantisoitu arvo on $Q_{\text{gyrox},v0} = 381$. Sama pätee kulmanopeusanturin molempien akselien ympäri olevien kulmanopeuksien mittaukseen, eli $Q_{\text{gyrox},v0} = Q_{\text{gyroy},v0}$. Arvot joita anturien tulisi näyttää paikallaan ollessa, sekä arvot joita ne näyttävät ja näiden erotus on esitetty taulukossa 3.1.

Taulukko 3.1. Mittaushetkellä laite on paikallaan ja vaakatasossa. Taulukossa arvot joita anturin tulisi olla ja arvot joita anturit näyttävät. Z-akseliin taas kohdistuu maan vetovoimasta aiheutuva kiihtyvyys 1 g, eli 9,81 m/s², jolloin anturilta tulevan signaalin jännite tulisi olla 1,65 V + 0,8 V = 2,45 V. Tällöin kvantisointitason $Q_{\text{accz},v0}$ tulisi olla 759.

	acc x	accy	acc z	gyro x	gyro y
Q_{v0}	511	511	759	381	381
Q	485	542	697	381	381
$Q_{v0} - Q$	26	-31	62	0	0

Osaltaan kiihtyvyyssanturin virhe selittyy sillä, että anturi on laitettu vain silmämääräisesti vaakatasoon, jolloin se voi olla vinossa ja putoamiskiihtyvyys vaikuttaa kaikkien kolmen akselin suuntaiseen mittaukseen. Anturi on kuitenkin mittaushetkellä paikallaan, jolloin kiihtyvyyksien summavektorin tulisi pituudeltaan olla putoamiskiihtyvyyden suuruinen, eli $Q_{\text{accSum},v0} = 759 - 511 = 248$. Summavektorin pituus lasketaan yhtälöllä

$$Q_{\text{accSum},v0} = \sqrt{(Q_{\text{accx}} - Q_{\text{accx},v0})^2 + (Q_{\text{accy}} - Q_{\text{accy},v0})^2 + (Q_{\text{accz}} - Q_{\text{accz},v0})^2} \quad (3.4)$$

Nyt summavektorin pituus on kuitenkin 190, eli virhe ei johdu vain siitä että anturi ei ole täysin vaakatasossa.

Kiihtyvyyden kvantisoidut arvot voidaan muuttaa m/s^2 asteikolle käyttämällä yhtälöä

$$a = 9,81 \text{ m/s}^2 / 0,8 \text{ V} * (V_{\text{ref}} * Q_{\text{acc}} / 1023 - 1,65 \text{ V}) \quad (3.5)$$

ja kulmanopeuden arvot $^\circ/\text{s}$ asteikolle yhtälöllä

$$w = 100 \text{ }^\circ/\text{s} / 1 \text{ V} * (V_{\text{ref}} * Q_{\text{gyro}} / 1023 - 1,23 \text{ V}) \quad (3.6)$$

Lopuksi muutetaan vielä kvantisoinnista saadut arvot binäärimuotoon. Binäärilukujärjestelmässä kantalukuna on 2, jolloin mikä tahansa positiivinen kokonaisluku voidaan esittää yhtälössä (3.6) olevaa, luvun kaksi potenssisarjaa käyttämällä. Kertoimet $b_n \dots b_0$ ovat bittijä, joista b_n on eniten merkitsevä ja b_0 vähiten. [12]

$$b_n * 2^n + b_{n-1} * 2^{n-1} + \dots + b_i * 2^i + \dots + b_1 * 2^1 + b_0 * 2^0 \quad (3.7)$$

10-bittisen A/D-muuntimen tapauksessa $n = 9$.

3.2 Suodattimen suunnittelu

Reaalimaailmassa mittaussignaalit sisältävät kohinaa, jotka aiheutuvat ympäristöstä, sekä anturien epäideaalisuudesta. Tätä kohinaa voi digitaalisessa signaalinkäsittelyssä tasoittaa keskiarvottamalla. Tämä tapahtuu summaamalla peräkkäisiä näytteitä haluttu määrä N , jonka jälkeen summa jaetaan näytteiden määrällä (3.8). [13]

$$x_{\text{ave}} = \frac{1}{N} \sum_{n=1}^N x(n) = \frac{x(1) + x(2) + x(3) + \dots + x(N)}{N} \quad (3.8)$$

Mitä enemmän peräkkäisiä näytteitä otetaan keskiarvon laskemiseen, eli mitä suurempi N on, sitä paremmin saadaan kohinaa suodatettua. Toisaalta jos suodatetaan liikaa, muuttuu pienien liikkeiden havaitseminen vaikeammaksi.

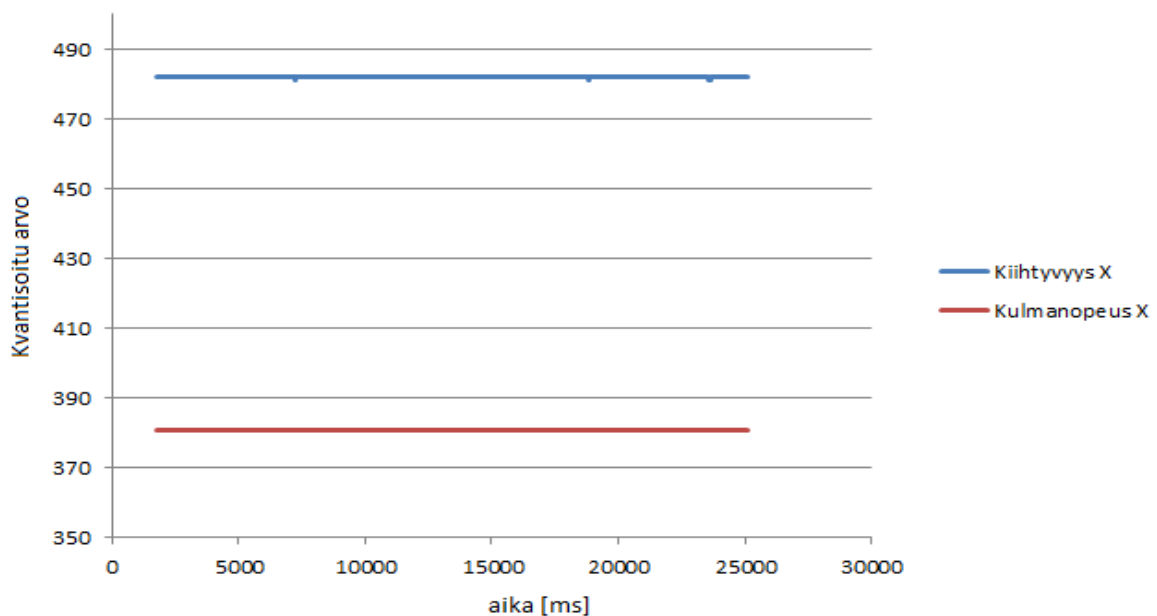
Oletetaan ennen testaamista, että summaamalla näytteitä 0,1 – 0,5 s ajalta saadaan kohinaa suodatettua, mutta silti havaitaan muutokset kiihtyvyydessä ja kulmanopeudessa. Näyteistystaajuuden tulisi olla vähintään $2 * f_{\text{MAX,gyro}}$, eli 280 Hz, jotta kulmanopeusanturilta tulevat signaalit eivät laskostuisi. Näytteitä tulisi siis ottaa vähintään 3,6 ms välein, jokaiselta viideltä käytössä olevalta mittauskanavalta ja näytteet täytyisi pitää tallessa aina halutulta ajalta Arduinin RAM:issa. Näytteet ovat tallennu int-tyyppisinä, kahden tavun kokoisina muuttujina. Tarvittavan muistin määrä tavuina on laskettu yhtälössä (3.9), missä t_N on ajan pituus, jona näytteitä summataan.

$$\text{Tarvittava muisti} = \frac{t_N}{1/f_{\text{MAX,gyro}}} * 5 * 2 \text{ tavua} \quad (3.9)$$

Kun $t_N = 0,1$ s, tarvittavan muistin määrä on 280 tavua ja vastaavasti kun $t_N = 0,5$ s, tarvittava muisti on 1400 tavua. Arduino Unossa on RAM:ia 2 kilotavua, mutta sinne tallennetaan myös muut ohjelmassa käytettävät muuttujat. GetFreeRam()-funktio ilmoittaa vapaaksi muistiksi 1278 tavua, kun näytteitä ei tallenneta ollenkaan. Eli 280 Hz:n näytteistystaajuudella pystytään laskemaan liukuvaa keskiarvoa maksimissaan 0,45 s ajalta muistin loppumisen takia.

Valitaan näytteistystaajuudeksi siten 300 Hz ja liukuvan keskiarvon laskemiseen otettavien näytteiden määräksi 100, jolloin tarvittava muisti on 1000 tavua. Nyt muistin loppuminen ei ole ongelma, eivätkä signaalit laskostu, koska $f_s > 2 * f_{\text{MAX,gyro}}$.

Kuvassa 3.2 on liukuvan keskiarvon suodattimella suodatetut samat signaalit kuin kuvassa 3.1. Nyt signaalien väre on huomattavasti pienempi, kuin ennen keskiarvottamista.



Kuva 3.2. Liukuvan keskiarvon suodattimella suodatetut kiihtyvyyss-, sekä kulmanopeussignaalit. Kun int-tyyppisillä muuttujilla lasketaan liukuvaa keskiarvoa pyörityvät tulokset aina alaspäin.

3.3 Nopeuden laskeminen

Kun ajetaan suoraan tasaisella nopeudella ja tasaisella alustalla, on mahdollista että kiihtyvyys ja kulmanopeus on kaikilla akseleilla 0. Tällöin näyttö lähtee päälle, mikäli ei trukin nopeutta mitata. Nopeuden mittaamista varten on kuitenkin ennen liikkeelle lähtöä kalibroitava anturit näyttämään arvoja, joita niiden paikallaan ollessa tulisi näyttää. Kalibrointi

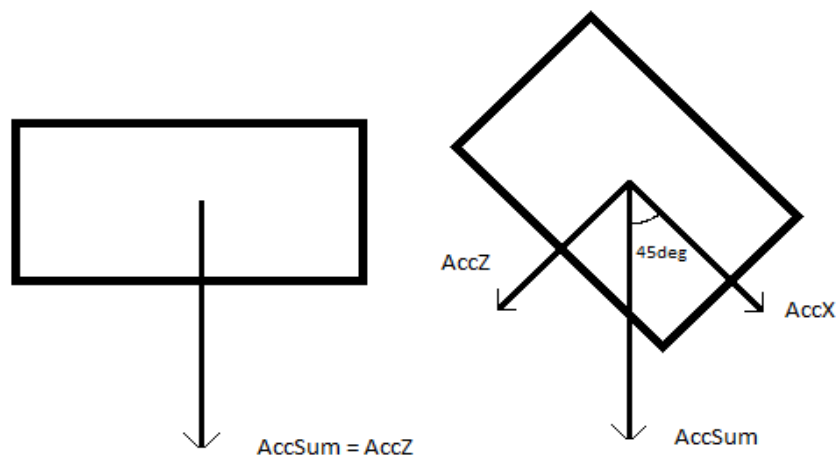
tehdään käynnistäessä ohjelma laskemalla nollakiihtyvyyden arvon ja anturilta tulevien arvojen keskiarvon erotus yhtälön (3.10) tavalla. Keskiarvo lasketaan 0,5 s ajalta näytteistystaajuuden ollessa 4000 Hz, jolloin $N = 2000$.

$$Q_{acc,Offset} = Q_{acc,v0} - \frac{1}{N} \sum_{n=1}^N Q_{acc}(n) \quad (3.10)$$

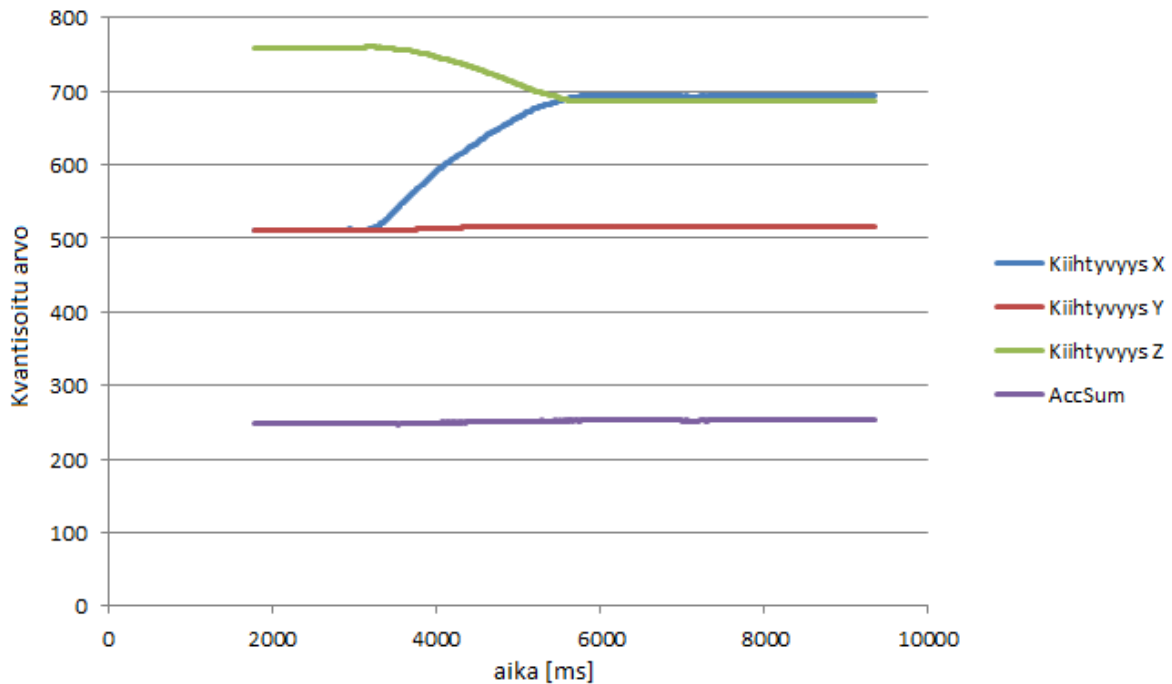
Tämä arvo lisätään suodatuksen jälkeen kiihtyvyyden arvoon, jolloin saadaan paikallaan ollessa $Q_{acc} = Q_{acc,v0}$. Sama tehdään kaikille kiihtyvyydsanturien akselleille, sekä kulmanopeusanturille. Nopeuden mittaaminen diskreetistä datasta tapahtuu yhtälön (3.11) mukaan summaamalla näytteiden arvoja ja kertomalla ajan pituudella, jona näytteitä summataan.

$$v = \frac{1}{N} \sum_{n=1}^N (Q_{acc}(n) + Q_{accx,Offset} - Q_{acc,v0}) * t_N \quad (3.11)$$

Nopeus voidaan laskea eteen – taakse kiihtyvyydestä, eli kiihtyvyydsanturien x-akselilta, mutta tämä on kuitenkin mahdollista vain silloin, kun alusta on tasainen. Kun anturi ei ole samalla tasolla kuin kalibroitaessa, aiheuttaa putoamiskiihtyvyys virhettä nopeuden mittaamiseen x-akselilta. Teoriassa putoamiskiihtyvyyden vaikutus nopeuden mittaukseen voidaan kuitenkin eliminoida laskemalla nopeutta kaikkien kolmen akselin summavektorilta. Summavektorin pituuden tulisi olla riippumaton siitä missä asennossa anturi on, kun ollaan paikallaan. Summavektorin laskenta oli esitetty yhtälössä (3.4). Testataan summavektorin riippumattomuus kulmaan, missä anturi on kääntämällä anturi 45 ° kulmaan siten että y-akselin suuntainen kiihtyvyys pysyy miltei vakiona. Kuva 3.3 havainnollistaa testiä ja kuvassa 3.4 on esitetty laitteen kääntämisestä aiheutuvat muutokset kiihtyvyydessä.



Kuva 3.3. Havainnollistetaan miten putoamiskiihtyvyydestä aiheutuvien kiihtyvyydsvektorien tulisi muuttua, kun anturi on eri asennoissa.



Kuva 3.4. Kuvassa esitetty kalibroidut ja suodatetut mittaussignaalit kaikilta kiihtyvyyksianturin akselilta, sekä kiihtyvyyksien summavektori, kun laitetta käännetään 45°. kuvan 3.3 mukaisella tavalla. Kiihtyvyyksianturin x-akselin muutos on $693 - 511 = 182$, y-akselin $516 - 511 = -5$ ja z-akselin $686 - 759 = -73$. Summavektorin arvo on alussa 248 ja lopussa 253, eli muutos on $248 - 253 = -5$.

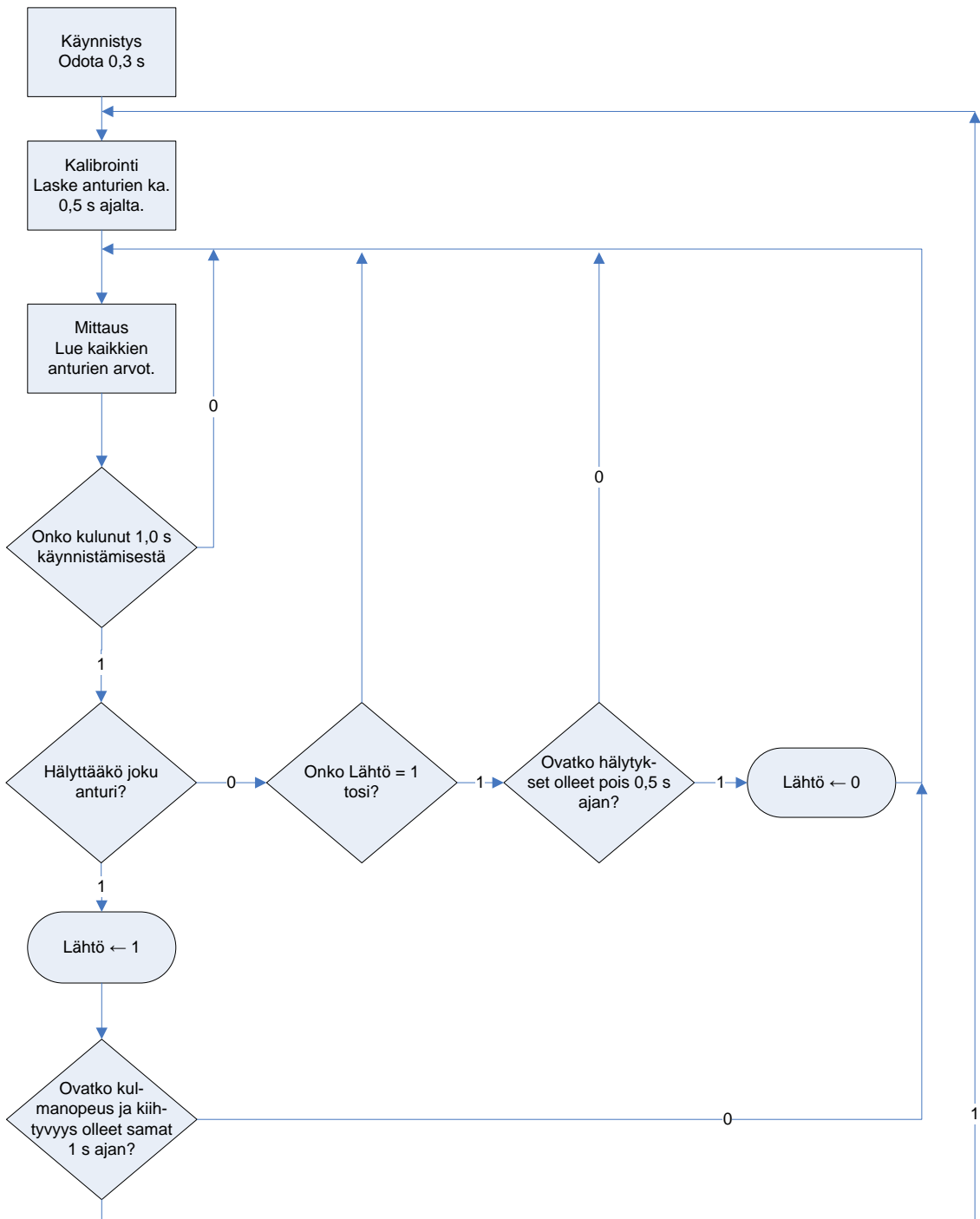
Kiihtyvyyksien summavektorin muutoksen tulisi olla 0, jotta voitaisiin olettaa, että pu-
toamiskiihtyvyyden vaikutus ajonopeuden laskemiseen epätasaisella alustalla pystytään
kumoamaan. Kuten kuvasta 3.4 huomataan, muutos kiihtyvyyksien summavektorissa on
kuitenkin yhtälöllä (3.4) laskettuna $0,2 \text{ m/s}^2$. Tämä muutos aiheuttaa mittaukseen virhettä,
jos ajetaan alustalle, joka on eri tasossa kuin kalibroitaessa. Tämä virhe tekee nopeuden
tarkasta mittaamisesta mahdotonta, jos alusta ei ole tasainen.

Joudutaan toteamaan, että kiihtyvyyksianturin epätarkkuuden takia nopeuden mittaaminen
on mahdotonta. Nyt kiihtyvyyksianturin y- ja z-akselien suuntaisten kiihtyvyyksien mittaaminen
on turhaa, koska niitä ei tarvittu muuhun kuin nopeuden mittaamiseen. Kulmanopeusantu-
rin y-akselin suuntainen mittaaminen on myös tarpeeton. Tämän takia aletaan käyttämään vain
kiihtyvyyksianturin eteen – taakse suuntaisen, eli x-akselin kiihtyvyyttä, sekä kulmanopeus-
anturin kääntymistä mittaavaa, eli x-akselin ympäri olevaa kulmanopeutta. Kokeile-
mällä havaitaan, että laskettaessa liukuvaa keskiarvoa 150 - 200 ms ajalta, anturi toimii
sopivalla herkkyydellä. Valitaan näytteiden määräksi keskiarvoa laskiessa 50, jolloin kes-
kiarvoa lasketaan 167 ms ajalta.

4. OHJELMOITU LOGIIKKA

Arduinon on koodattu suodattimen ja hälytysrajojen lisäksi logiikka, miten laite toimii. Tilakaavio logiikasta on esitetty kuvassa 4.1. Kun laite käynnistetään, odotetaan 0,3 s ennen kuin aletaan lukea antureilta tulevia signaaleja, jotta anturit ehtivät käynnistyä. Tämän jälkeen kalibroidaan anturit edellisessä kappaleessa esitetyllä tavalla. Aikaa on nyt kulunut 0,8 s käynnistymisestä ja aletaan laskemaan liukuvaa keskiarvoa. Keskiarvon laskemiseen käytetään 50 näytettä, joiden ottamiseen kestää 300 Hz taajuudella 0,167 s. Tämän takia anturien arvojen vertaaminen hälytysrajoihin aloitetaan vasta, kun ohjelma on ollut käynnissä 1,0 s.

Nyt uuden näytteen tullessa tutkitaan, että ylittääkö jompikumpi anturien liukuvista keskiarvoista hälytysrajan. Kun ylitys tapahtuu, sijoitetaan lähtöön 1, jolloin summeri alkaa soimaan. Anturien arvojen pitää olla hälytysrajojen välissä 0,5 s yhtäjaksoisesti ennen kuin lähtöön laitetaan 0, jolloin summeri lopettaa soimisen. Tällä tavalla pyritään estämään hälytyksen ”välkkyminen” ajaessa. Kun ollaan hälytysrajojen ulkopuolella, tutkitaan myös että pysyvätkö anturien arvot muuttumattomina. Tämä tehdään siksi että jos ajetaan kaltevuudeltaan erilaiselle alustalle, kiihtyvyyssanturin arvo on putoamiskiihtyvyydestä johtuen hälytysrajojen ulkopuolella vaikka ollaan paikallaan. Kiihtyvyys pysyy kuitenkin paikallaan ollessa samana. Jos kiihtyvyyden ja kulmanopeuden arvoissa ei ole muutosta hälytystilassa ollessa 1 s aikana, kalibroidaan anturi uudelleen. Nyt hälytys lähtee pois päältä, kun hälytysrajat eivät ylity 0,5 s ajan. Eli yhteensä aikaa hälytyksen loppumiseen kuluu 2,0 s siitä hetkestä kun anturien arvot ovat muuttumattomat.

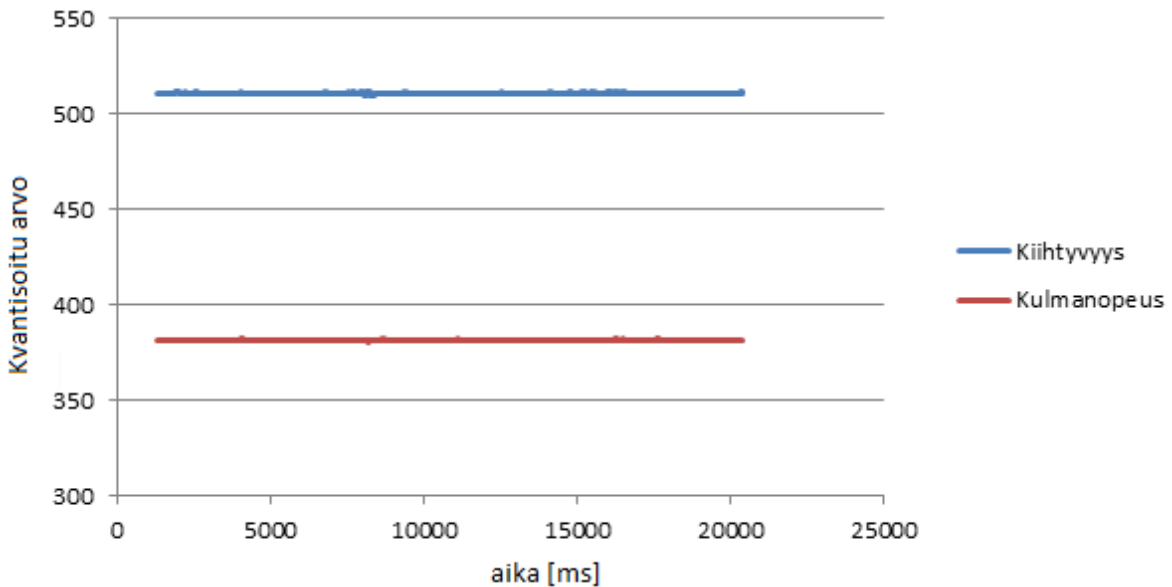


Kuva 4.1 Kaavio logiikan toiminnasta. Suorakaiteen muotoiset ovat tilaelementtejä, salmiakin muotoiset päätöksentekoelementtejä ja soikio on ehdollista toimintaa kuvaava elementti. Päätöksentekoelementeillä on yksi tulo ja kaksi lähtöä, toinen on 0, eli epätosi ja toinen 1, eli tosi.

5. TESTIT JA MITTAUKSET

Testit suoritettiin henkilöautolla, koska testattaessa ei ollut trukkia käytössä. Suoritettiin neljä erilaista testiä, joilla osoitetaan liiketunnistimen toimivuus eri tilanteissa. Ensimmäisessä käynnistetään auto, käytetään tyhjäkäynnillä ja sammutetaan. Tämän testin tarkoitus on saada selville kuinka paljon anturien auton tyhjäkäynnistä aiheutuva värinä vaikuttaa anturien näyttämiin arvoihin, jolloin voidaan määrittää hälytysrajat antureille. Toisessa testissä kiihdytetään, ajetaan tasaisesti suoraan eteenpäin ja pysähdytään, sekä peruutetaan takaisin lähtöpisteeseen. Kolmannessa testissä ajetaan niin kuin trukilla oletetaan ajettavan, eli mutkitellaan, pysähdellään ja peruutetaan. Viimeisessä testissä ajetaan tasaiselta alustalta ylämäkeen, jolloin saadaan selville kalibroinnin toiminta, kun pysähdytään kaltevuudeltaan kalibrointitasosta poikkeavalle alustalle.

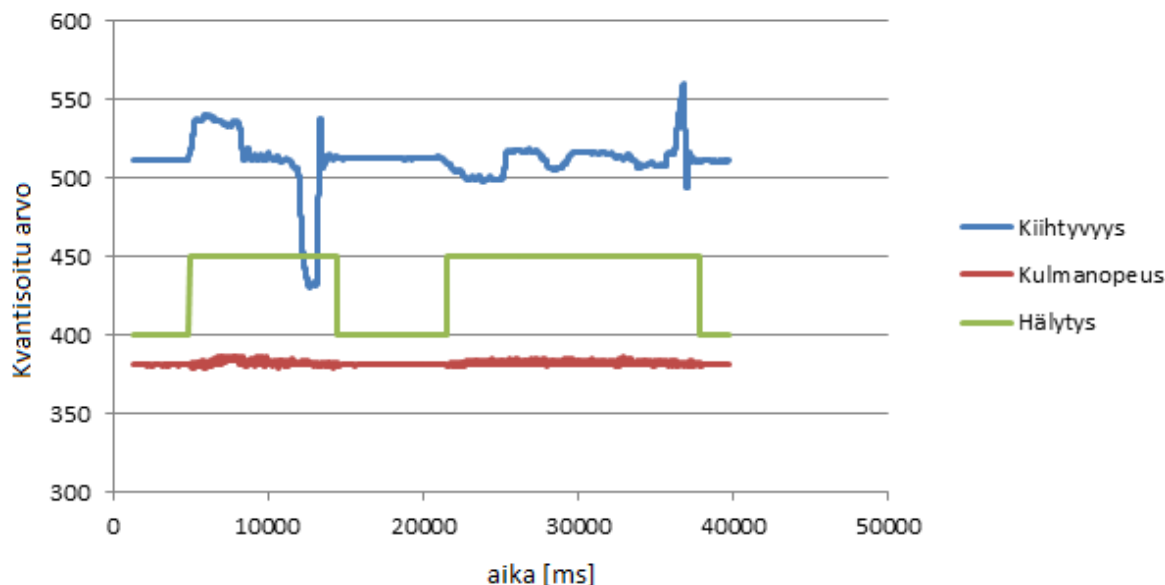
Kuvassa 5.1 on esitetty ensimmäisen testin datoista piirretty kuvaaja. Moottorin käynnistäminen tapahtuu ajanhetkellä 8 s ja sammuttaminen 15 s.



Kuva 5.1. Suodatetut kiihtyvyyden ja kulmanopeuden arvot. Näytteistystaajuus 300 Hz ja liukuvaa keskiarvoa lasketaan 0,15 s ajalta, kuten muissakin tämän kappaleen testeissä.

Kuvassa 5.1 esitetyn kiihtyvyyden arvo vaihtelee välillä 510 – 512 ja kulmanopeuden arvo välillä 380 – 381. Hälytysrajoja ei laiteta näihin arvoihin, vaan jätetään hieman varaa, jolla varmistetaan ettei liiketunnistin hälytä paikallaan ollessa. Määritetään kiihtyvyyden hälytysrajoiksi 514 ja 508, eli yhtälöllä (3.5) muunnettuna $0,14 \text{ m/s}^2$ ja $-0,14 \text{ m/s}^2$. Kun $Q_{\text{accx}} > 514$, tai $Q_{\text{accx}} < 508$ oletetaan että ollaan liikkeessä. Kulmanopeusanturin vastaaviksi rajoiksi asetetaan $Q_{\text{gyrox}} > 383$ ja $Q_{\text{gyrox}} < 379$, jotka ovat yhtälöllä (3.6) muunnettuna $0,6 \text{ }^\circ/\text{s}$ ja $-0,6 \text{ }^\circ/\text{s}$.

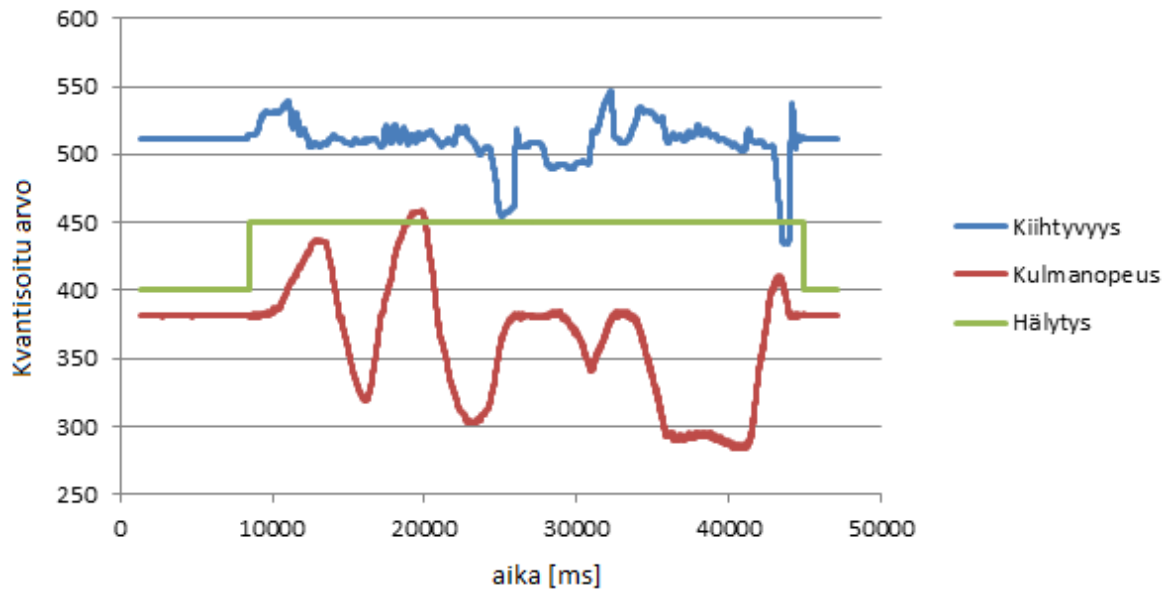
Toisessa testissä käytetään määritettyjä hälytysrajoja ja testataan laitteen toiminta, kun kiihdytetään noin 10 km/h nopeuteen, ajetaan suoraan tasaista nopeutta ja pysähdytään. Tämän jälkeen ollaan paikallaan ja peruutetaan takaisin lähtöpisteeseen. Kuvassa 5.2 on esitetty testissä tallennettu data.



Kuva 5.2. Toisen testin datoista piirretyt kuvaajat. Kuvassa myös digitaalisen ulostulon 13 lähtö, joka todellisuudessa saa arvoja väliltä 0 – 1, mutta arvot on nyt skaalattu välille 400 – 450, jotta ne näkyvät kuvassa paremmin.

Liikkeelle lähdetään ajassa 5 s, jolloin kiihtyvyyden kuvaajassa voidaan havaita kasvu välillä 5 – 9 s. Ajassa 5 s hälytys lähtee myös päälle. Välillä 9 – 11,5 s ajetaan tasaisella nopeudella, jolloin kiihtyvyyden arvo vaihtelee välillä 510 – 515 johtuen siitä että nopeus ei pysy aivan tasaisena, ja siitä että alusta ei ole aivan tasainen. Tämän jälkeen jarrutetaan välillä 11,5 – 13,5 s, jolloin kuvassa on havaittavissa negatiivinen kiihtyvyys. Noin sekunnin kuluttua pysähtymisestä hälytys lähtee pois päältä, kunnes se nousee taas päälle ajassa 21,5 s, kun aletaan peruuttamaan. Peruuttaminen tapahtuu aikavälillä 21,5 – 37 s, jonka jälkeen taas ollaan pysähdyksissä ja hälytys lähtee pois päältä ajassa 38 s. Kulmanopeuden ei tulisi muuttua ajettaessa suoraan, mutta testattaessa siinä havaitaan muutos aina liikkeellä oltaessa.

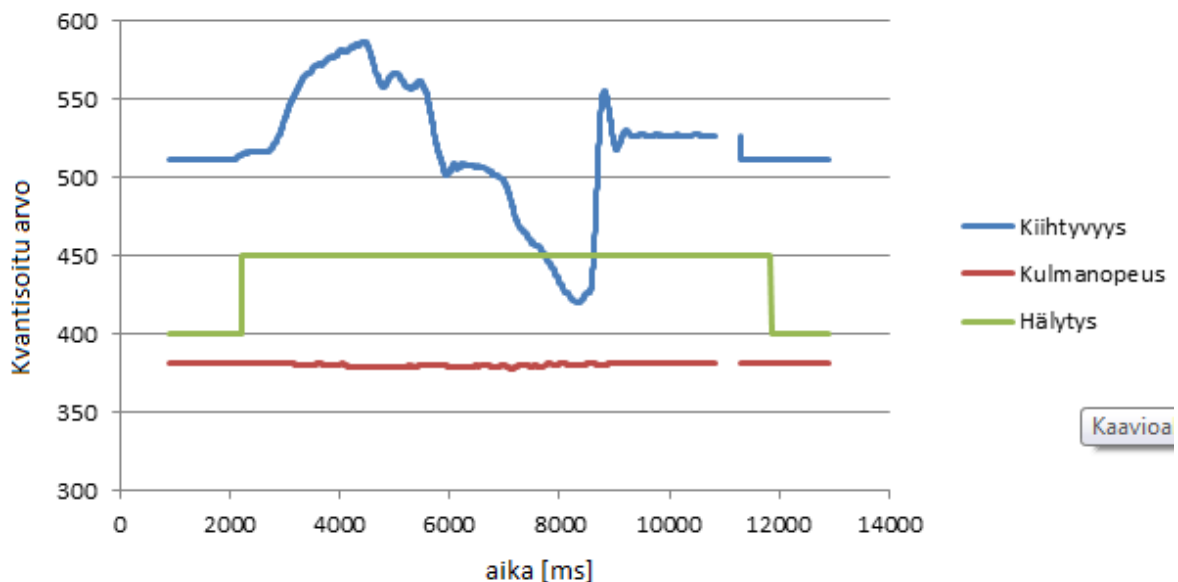
Kolmannessa testissä yritetään havainnollistaa laitteen toimintaa, kun ajetaan samalla tapaa kuin trukeilla usein ajetaan. Tämä tehdään kääntyilemällä, sekä tekemällä lyhyitä pysähdyksiä, joiden jälkeen vaihdetaan ajosuuntaa. Taltioiduista datoista piirretyt kuvaajat on esitetty kuvassa 5.3



Kuva 5.3. Trukille ominaisella ajotavalla ajetun testin kuvaajat.

Kuvasta huomataan että ajassa 8,5 s tapahtuvan liikkeelle lähdön jälkeen hälytys pysyy päällä aina aikaan 45 s asti jolloin pysähdytään.

Testataan vielä logiikan toiminta pysähdyttäessä kaltevuudeltaan lähtöpaikasta olevalle tasolle. Kun anturien arvot pysyvät muuttumattomina 1 s ajan, kalibroidaan anturi. Kuvassa 5.4 on esitetty testissä taltioiduista datoista piirretyt kuvaajat.



Kaavioa

Kuva 5.1. Kalibroinnin testaus. Kiihtyvyys ja kulmanopeus signaaleissa ajassa 10,8 s -11,3 s oleva katkos johtuu silloin tapahtuvasta kalibroinnista.

Liikkeelle lähdetään ajassa 2,2 s, jolloin hälytys lähtee päälle. Tämän jälkeen kiihdytetään ja jarrutetaan, kunnes ajassa 9,4 s ollaan taas paikallaan. Alustan kaltevuudesta johtuen kiihtyvyyden arvo ei olekaan 511, vaan 527, jolloin hälytys jää päälle. Ajassa 9,8 s on kiihtyvyyden ja kulmanopeuden arvot muuttuneet lähes vakioiksi, jolloin odotetaan 1 s ajan ja tämän jälkeen kalibroidaan anturi. Kalibrointiin kuluu noin 0,5 s aikaa, jonka jälkeen odotetaan vielä että hälytysrajat eivät ylity 0,5 s ajan. Nyt hälytys lähtee pois päältä ajassa 11,8 s. Pysähtymisestä hälytyksen sulkemiseen kului aikaa $11,8 \text{ s} - 9,4 \text{ s} = 2,4 \text{ s}$.

6. YHTEENVETO

Tavoitteena oli suunnitella ja toteuttaa laite, jolla pystytään havaitsemaan milloin trukki liikkuu ja tällöin sulkea trukin tietokoneen näyttö työturvallisuuden parantamiseksi. Käytettiin liikkeen havaitsemiseen MEMS tekniikalla toteutettuja kiihtyvyyss- ja kulmanopeusantureita. Antureita luetaan Arduino Uno mikroprosessorikehitysalustalla, johon ohjelmoidun logiikan perusteella kytketään testivaiheessa näytön paikalla oleva summeri päälle ja pois.

Liiketunnistin saatiin toimimaan johdannossa esitetyn kuvan 1.1 mukaan muuten paitsi nopeuden mittauksen osalta. Nopeuden tarkan mittauksen estää kiihtyvyyssanturin mittausakselien epäideaalisuudesta johtuva jatkuvuustilan virhe, joka syntyy kun ajetaan epätasaisella alustalla. Käytettyjen analogisten anturien tarkkuus on riittävä liikkeen havaitsemiseen. Suodattamalla signaaleja ensin analogisesti ja sitten digitaalisesti saadaan mitaussignaalien kohina tarpeeksi pieneksi, ettei se häiritse liiketunnistusta. Ohjelmoidun logiikan avulla saatiin ”välke” summerin soimisessa poistettua, sekä estettyä se että summeri jää soimaan pysähdyttäessä kaltevuudeltaan erilaiselle alustalle. Nopeuden mittauksen puuttuminen ei ole kuitenkaan välttämättä haitallista, sillä trukilla ei oletettavasti ajeta pitkiä aikoja tasaisella nopeudella suoraan eteenpäin, joten trukin tietokonetta ei pysty ajon aikana käyttämään. Kuten kappaleen 4 toisesta testistä havaitaan, tilanteeseen jossa ollaan liikkeellä, mutta hälytys lähtee pois päältä, ei päästä vaikka yritetään ajaa suoraan tasaista nopeutta, koska anturit ovat niin herkkiä että pienetkin muutokset alustassa, nopeudessa tai ohjauskulmassa havaitaan. Jotta nopeuden luotettava mittaus saataisiin tehtyä, olisi kokeiltava tarkempaa kiihtyvyyssanturia, tai kokonaan muuta tekniikkaa.

Testaamista ei kuitenkaan päästy tekemään trukilla vaan testit suoritettiin henkilöautolla. Trukin liikkeen tunnistamiseen erona henkilöautoon on moottorin aiheuttama värinä, joka on erilainen riippuen onko trukissa poltto- vai sähkömoottori. Tämä ongelma voidaan todennäköisesti poistaa sopivalla suodatuksella. Trukeissa voi ongelmaa aiheuttaa myös se kun nostetaan painavia kuormia, niin trukin eturenkaat voivat painua lyttyyn, jolloin kiihtyvyydessä havaitaan muutosta putoamiskiihtyvyyden takia, vaikka trukki olisikin paikallaan. Prototyypin mukaisia liiketunnistimia on asennettu trukkeihin toimintaan kolme kappaletta syksyllä 2011. Laitteet toimivat spesifikaation mukaisesti ja kenttätetit jatkuvat.

LÄHTEET

- [1] O'Neal, C. B., Malshe, A. P., Singh, S. B., Brown, W. D., 1999, Challenges in the packaging of MEMS, Julkaistu: Advanced Packaging Materials: Processes, Properties and Interfaces, 1999. Proceedings. International Symposium on, IEEE, s. 41 - 47
- [2] Je, H. C., Lee, S., Lee, M. L., Lee, J., Yang, W. S., Choi, C. A., 2010, Z-axis Capacitive MEMS Accelerometer with Moving Ground Masses, NT Convergence Components Research Department, Electronics and Telecommunications Research Institute (ETRI), Daejeon, Korea, IEEE, s. 635 - 638
- [3] Freescale Semiconductor 2008. Technical data. MMA7361L. [verkkodokumentti]. [Viitattu 28.09.2011]. Saatavilla http://www.freescale.com/files/sensors/doc/data_sheet/MMA7361L.pdf
- [4] Sparkfun tuotekatalogi. Triple Axis Accelerometer Breakout - MMA7361. [verkkodokumentti]. [Viitattu 28.09.2011]. Saatavilla <http://www.sparkfun.com/products/9652>
- [5] STMicroelectronics. Technical data. LPR510. [verkkodokumentti]. [Viitattu 31.10.2011]. Saatavilla <http://www.pololu.com/file/0J239/lpy510al.pdf>
- [6] Shkel A. M., 2006, Type I and Type II Micromachined Vibratory Gyroscopes, Julkaistu: Position, Location And Navigation Symposium, 2006, IEEE/ION, s. 586 - 593
- [7] TKT-2540 Paikannuksen menetelmät. [verkkodokumentti]. [Viitattu 20.12.2011]. Saatavilla <http://www.tkt.cs.tut.fi/kurssit/2540/paikannuspruju.pdf>
- [8] Polulo tuotekatalogi. LPR510AL Dual-Axis (Pitch and Roll or XY) Gyro with ± 100 %/s and ± 400 %/s Ranges. [verkkodokumentti]. [Viitattu 31.10.2011]. Saatavilla <http://www.pololu.com/catalog/product/1266>
- [9] Arduino kotisivu. Hardware. Arduino Uno. [verkkodokumentti]. [Viitattu 31.10.2011]. Saatavilla <http://arduino.cc/en/Main/ArduinoBoardUno>
- [10] FAIRCHILD Semiconductor. Technical data. BC547. [verkkodokumentti]. [Viitattu 07.11.2011]. Saatavilla <http://www.datasheetcatalog.org/datasheet/fairchild/BC547.pdf>
- [11] Sedra, A., Smith, K. Microelectronic Circuits. 5th ed. New York, 2004. Oxford University Press. s. 419-421.
- [12] Bentley, J. Principles of measurement systems. Singapore 1985. Longman Singapore Publishers. s. 209-215
- [13] Lyons, R. Understanding Digital Signal Processing. 3rd ed. Ann Arbor, Michigan 2011. Edwards Brothers. s. 589